Hypermedia and the Textbook
Teaching C++

Curtis Sollohub
New Mexico Highlands Unviersity

I. The Late Age of Print

In his book, Writing Space, Jay David Bolter (1991) talks about "the late age of print." He argues that we will soon witness the end of the book as a bound collection of paper pages with a preset beginning and end. What will take its place will be the electronic book, without a necessarily fixed content, order, or boundary. In the electronic book, the content can be modified by author or even by reader as needs be, the order will depend on how the reader chooses to follow a set of hypertext links (an extension of the links we see in Internet web documents), and the boundaries will be shattered by hypertext links that reach beyond the immediate document under consideration. This paper discusses the implications of this notion with regards to the textbook and presents the rudimentary beginnings of such a text to teach the programming language C++.

The traditional printed textbook emphasizes the author as expert and the reader as passive recipient of that expertise. In such a book, the author's expertise often shines through and the instructor can get excited about the possibility of using the product of the author's efforts. At the same time, given the ways of professorship, the instructor usually finds aspects of the text that are not 'quite right' and wishes that the good could be borrowed while the inappropriate is discarded.

The student has a different experience of such a text. Ironically, the author's expertise often gets in the way of student usage of the material. The student reader is confronted with a hardbound volume, with well-laid out, authoritative text and figures. Such a book becomes something to revere, not something to read critically and mark up, let alone modify.

Computer-based hypermedia changes the relationship between writer and reader. The nature of the text almost requires that the reader be an active participant, as reading becomes an exploration. Each click of the mouse to follow a link is a decision by the reader. The more links, the more decisions. The reader can choose to:

- follow along in the physical order laid out by the author
- review already explored material
- consider an example
- explore an alternative explanation
- skip to a summary
- call up a tutorial
- send email to the instructor.

On a more practical level, a section can be printed out, marked up, and even modified as the student struggles to understand the material and rephrase it in his or her own words.

For instructors, such a text can easily become a collaborative effort, with each instructor choosing which sections to accept as a whole, which to modify, and which to completely rewrite. The 'same' text can have multiple paths as instructor-authors add their own contributions.

As envisioned here, such a text would be delivered via some kind of computer. In the future, this computer would be a variation of the laptop, with wireless access to a 'library' containing any number of ‘books’ available for downloading. Such technology is not generally available yet and there are serious challenges to overcome. One of these is the fact that with the present technology the average person takes in computer text 25% slower than text on a page. Likewise, reading from a screen seems to cause more eyestrain.

There is a further, perhaps more important issue. If one accepts the notion that “the media is the message” and its implication that each type of media makes it easier to present certain ‘truths’ and harder to present others, will the hypermedia text change the very nature of what material is presented and how it is presented? To explore these questions, the author of this paper has written a set of web-based documents to teach C++. The materials include a ‘narrative’ on C++, reference materials, programming patterns, and examples. He is presently working on tutorials on key topics. All of this is available on the web at the following address: http://cs.nmhu.edu/personal/curtis/cs1htmlfiles/Cplusintro.htm

The fall of 1998 represents the third semester that this material has been used in the classroom. Results so far indicate both the promise and the problems with such material. The next section describes the text and the third section describes some of the preliminary results.

II. Teaching C++ in the Late Age of Print

Students come to a text with different goals. Some already know at least part of the material and want to start in the middle. Others are looking for a reference document, especially if they have studied the material before. Some know that they learn best from examples and prefer to start with such and move to more detailed explanations later. Still others need a carefully crafted guide with plenty of ‘story’ and metaphor.

This last group is really many groups. A ‘guidebook’ can be written in many different ways, with each way preferred by a different type of student. To what extent should pictures and diagrams be included? How should the topics be ordered? At what point should key abstractions be introduced and how thoroughly? What level of mathematics should be included in the discussion? What types of metaphors and analogies should be used? What learning styles (visual, tactile, or auditory) should be emphasized and how? No single print-oriented text can possibly meet the needs or expectations of all its readers. It does seem, however, that attempts have been made to do just that, and elsewhere (http://cs.nmhu.edu/personal/curtis/txtconcp.htm), the author has argued that the cumbersome length of some C++ texts can be explained by attempts to meet all these needs and expectations.

It is important to understand that what is being advocated here is not a specific pedagogy. The point is not to follow, for example, a problem- or case-based approach. Nor is this a discussion about the importance of rigor, about which programming language to teach, or about whether to present an object-based or functional approach to programming. Rather, it concerns the need to sort out the various demands of a textbook and give each aspect its due.

To that end, the material to teach C++ described here is really a set of distinct documents, connected internally and externally via links. The first of these has the working title "The Story of C++." At present, this is the longest of the documents. It acts as a narrative description of the material in C++ and has the flavor of a traditional textbook except that its chapters and sections are connected via web links. Such links also connect this material to the other documents in this set.

A second document goes under the working title “The Essentials of C++.” This represents an easy to read manual of the basic constructs of the language. Again, each section contains links to other sections and other documents.

A third key document is a set of examples that one could read through as a text. Many of the examples include just enough explanation that a reader with good programming skills and a knack for learning from examples could use this material as the primary learning tool.

This last document is not complete at this writing. However, even as it presently stands, it represents one of the many potential routes being developed for student exploration. Other examples include:

- a set of on-line tutorials using an off the shelf computer-based training (CBT) development tool, Asymetrix

- animations and game-based approaches

- audio explanations.

While some documents are more finished than others, it is the nature of hypertext that none of the documents are truly completed. For example, a complete re-write of one of the chapters of "The Story of C++" is being completed now. It is envisioned not as a replacement for the present chapter but as an alternative reading. At present, students can either read one or both of the versions. Later, after it has been determined which approach works better with which type of student, the two variations could be preceded by a description that allows students to choose which approach they want to start with.

III. Preliminary Results

Students seem to like the web-based approach described here but there is a learning curve to its effective use. Many students seem to end up reading "The Story of C++" while ignoring the rest of the material. This is similar to reading a regular text while ignoring all the appendices and other supporting materials. To counter this practice, this semester the author will include discussion of how to use the material in the lab sessions.

At this point, most students still buy a printed copy of the materials. In the first versions of the material there was little difference in content between the on-line and printed versions; the main difference was the links. Now, however, as more material has been added that is not being printed and as the web of links has grown, there is a significant difference in the two products. Students need to understand what the on-line version provides and, again, education is the answer. In addition, to accommodate students who do not have web access at home or wherever they use a computer, a CD-ROM version of the complete set of documents is being made available.

Because the on-line version is free while the printed version costs, the Computer Science department has been forced to make a rule prohibiting printing of the text in the lab. Most students respect this and don't complain. A few, however, have simply gone to a computer lab outside the Computer Science department and printed as much of the text as they wanted.

HTML is a limited form of hypertext and as such does not allow for complete exploration of the medium. For example, it has proven difficult to link in quality CBT modules and high quality graphics. Likewise, one does not have sophisticated control over the presentation of links. On the reader's side, it is not easy get an overview of the document web. Student's traversing the space often desire such a 'map'. Other hypertext products, such as "StorySpace," (http://www.eastgate.com/Storyspace.html), provide these capabilities and are actively being considered for future iterations of this material.

Advantages for the on-line version include the following:

1. The student can cut and paste any piece of code found in the material directly into a program editor. This encourages students to borrow and explore code as opposed to simply reading it.

2. The same cut-and-paste capability allows students to create their own set of notes by cutting out key material and mixing it with their own rephrasings. One way to encourage this rephrasing is by requiring that students submit electronic journal entries involving the rephrasing of the material. This can be seen as a logical extension of the notion of a book without boundaries.

3. The material is available anywhere one has access to a networked computer. Students in the lab can access what they are looking for even if they have forgotten their text.

4. Since web browsers include document search capabilities, it is possible to locate references to specific concepts whether or not they are included in an index.

IV. Summary

No quantitative studies have been completed yet on how well students perform using this material. It is not complete enough yet to do so. The author is personally focusing on adding tutorial modules and on designing high-quality graphics as explanations of abstract and conceptually difficult material. He is also actively recruiting other instructors of C++ to participate in this effort. Anyone interested should contact Curtis Sollohub at curtis@cs.nmhu.edu. To explore the material described here, go to: “http://cs.nmhu.edu/personal/curtis/cs1htmlfiles/cs1text.htm”.

References

1. Bolter, David Jay (1991). *Writing Space*. Hillsdale, N.J.: Lawrence Erlbaum.

2. Sollohub, Curtis (1997). The Computer Science I Textbook as a Web Document. *The Journal of Computing in Small Colleges* (13)2: 179-184.